Analysis Of Job Scheduling Algorithm For An E-Business Model In A Cloud Computing Environment Via GI/G/3/N/K Queuing Model

Dr Riktesh Srivastava
Assistant Professor, Information Systems
Skyline University College
University City of Sharjah
SHARJAH-UAE
[rsrivastava@skylineuniversity.com]

Abstract

Conventional e-Business models using the Client/Server architecture has always been complex, thorny and expensive. The quantity and assortment of hardware, software and middleware technologies used in order to run them was intimidating. With the emergence of Cloud computing, the technologies needed for an e-Businesses application are managed by experienced third parties who offer servers, storage management, network technologies, and virtualization technologies. This manuscript presents the accomplishment of the job scheduling algorithm developed for an emblematic e-Business model using GI/G/3/n/k queuing theory perception. Finally, the associated simulations and numeral results are provided, to assurance the Quality of Service (QoS) for all throughput of system.

Keywords: Cloud computing, GI/G/3/n/k Queuing theory, QoS, job scheduling algorithm.

1. Introduction

Advent of Internet has changed the mode conventional businesses were conducted. Nevertheless, with the augmentation of Internet, the magnitude of users has also amplified significantly. E-Marketer forecasts that B2C e-commerce sales in Asia pacific region has grown by 23.3% annually and reached $168.7 billion by 2011 [1]. According to [2], there is escalation of 444.8% usage of Internet users in 2010-2011 itself. An E-Business model, predominantly, B2C E-Commerce, lessens communication costs by escalating consumer admittance to information and permitting consumers to trace the most doable value for the product. B2C e-commerce also ease market entry barriers as the price of developing and maintaining a Web site is much cheaper than the traditional business structure. However, it augments the network traffic on e-Business model, ensuing in trouncing of requests send by the consumers and impasse of the queue at any stage of the complete architecture. Many organizations suffered losses due to slow performance of Internet servers and were looking for the solution to manage the increased traffic. The solutions to these problems were found in ***Cloud Computing***. Cloud Computing is a computing paradigm, where large pools of systems are connected in private or public networks, to provide dynamically scalable infrastructure for application, data and file storage. With the introduction of this technology, the cost of computation, application hosting, content storage and delivery is reduced significantly [3].

In this research, a job scheduling algorithm is proposed which aspires to significantly minimize the Service time of the user’s requests, thereby improving the overall QoS of the e-Business

architecture being implemented using Cloud computing environment. The algorithm is based on ***GI/G/3/n/k*** queuing model, where GI symbolizes general independent arrivals, G represents general (any distribution) service, 3 defines the number of servers (web server, application server and database server) used in the proposed architecture, n is the capacity of the queue (capacity of the queue in the research is summation of buffer size at servers) and k represents the number of requests arriving at web server of the projected architecture.
In the research, the software code is developed using Java 2.0 which primary tests the implementation of the projected architecture on the conventional Client Server environment. The same architecture is then experienced on Cloud Computing scenario and the comparative study depicts that the Service time of the requests in the later is much faster, effective and efficient.

The rest of the paper is divided as follows: Section 2 elaborates the traditional 3-tier architecture e-Business model and its implementation in Client Server environment along with the in the mathematical assumptions for the same. Section 3 elaborates diverse distribution techniques which are used in the projected Job scheduling algorithm. The section also mentions the algorithms of 5 different distributions that form General distribution. Section 4 depicts the step by step execution of the Job scheduling algorithm using flowcharts of formation of arrival instances, formation of departure instances and service time evaluation respectively. Section 5 reveals the results and analysis of the algorithm with the comparative study between M/M/1 and GI/G/3/n/k queuing models. Section 6 proposes the e-Business architecture in a Cloud Computing environment. The section also mentions the step by step procedure of the proposed architecture. Section 7 executes the Job Scheduling algorithm in the proposed architecture and evaluates the results using Regression technique, thereby, formulating the mathematical equation of the Service time. Section 8 does the performance analysis of the proposed algorithm in both the e-Business environment and mentions the findings. Section 9 concludes the paper.

2. e-Business architecture in the Client Server environment

The early e-Business architecture was based on 2-tier configuration, as specified in Figure 1.

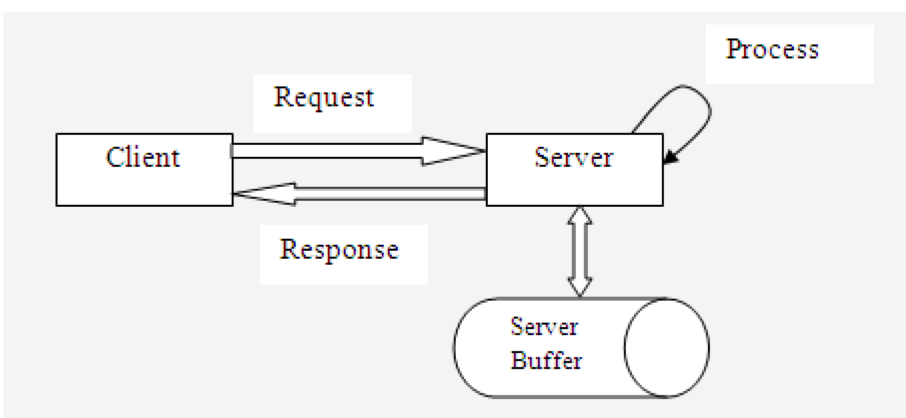

Figure 1: 2-Tier e-Business architecture

The model mentioned in Figure 1, was well recognized and adopted in business, being more competent by distributing processing among client and server. In the said model, the Client sends request to the server, which gets processed and response is send back to the Client. The quandary arises, when the number of clients' increases, the request has to indefinitely wait in the Server

Buffer. To remove this problem, 3-Tier e-Business architecture was proposed as indicated in Figure 2. It must be noted that in 3-Tier EC architecture, the number of Application Server can be more than 2 (up to n, depends on the complexity of the system).

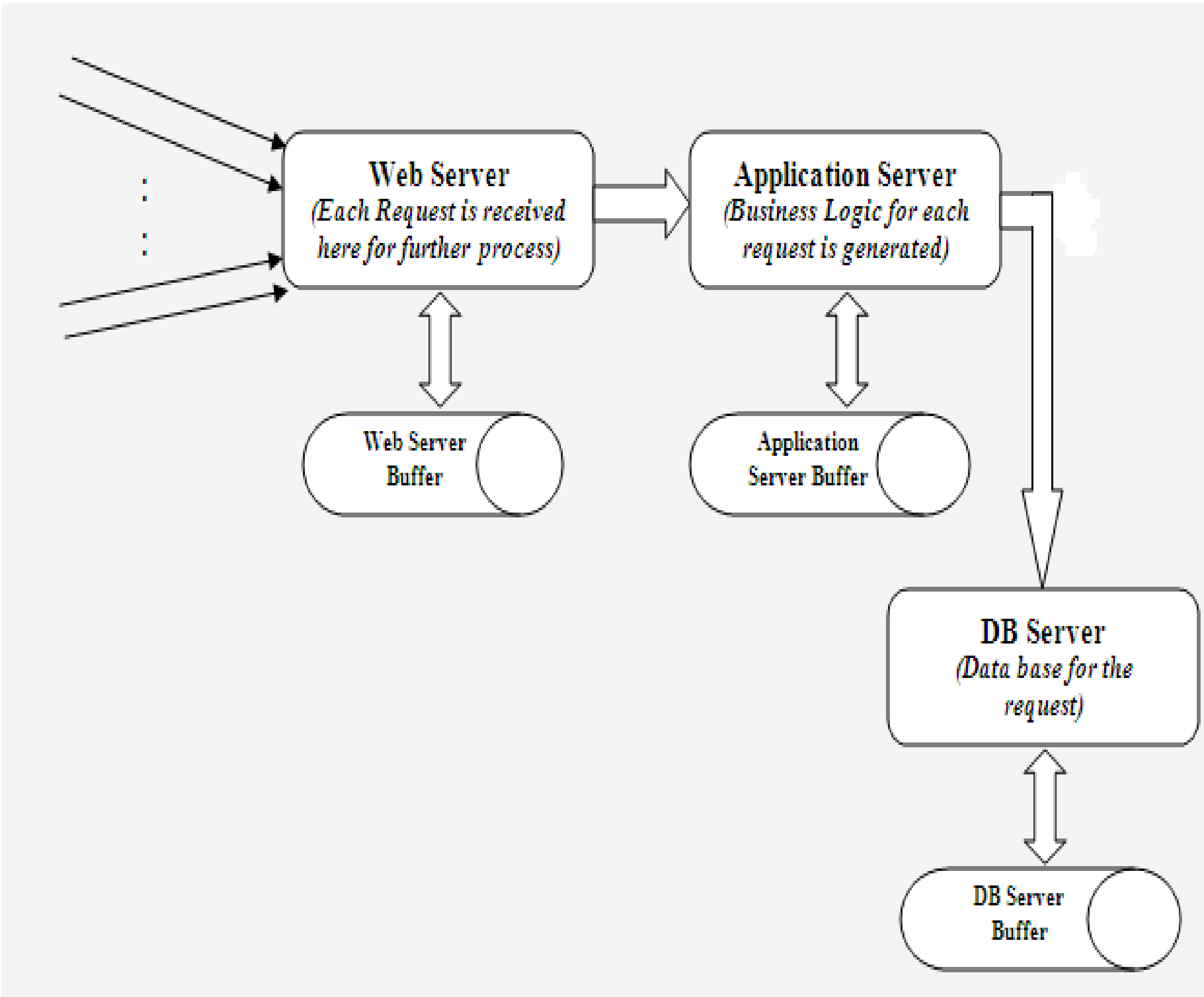

Figure 2: 3-Tier e-Business architecture

For simplicity, a hypothesis that entire development is abridged to a single arrival process and the response is engendered consequently at apiece tier of the e-Business architecture. In the study, all the request and response generated are random; the assessment of throughput is a foremost challenge as:

1. *No data ought to be overflowed / freezed out.*
2. *The entire architecture must guarantee for its unwavering function.*

These two problems are of foremost substance of the planned study. If we scrutinize Figure 2, the e-Business architecture bear a resemblance to Queuing theory, it is therefore premeditated to lug out the study by employing Queuing Theory.

2.1 Mathematical Assumptions for 3-Tier e-Business architecture

It is assumed that the number of requests that disembark at the web server are $\left[R_1, R_2, ..., R_n\right]$. These requests are random in nature and are being denoted by λ. Based on these assumptions, the total number of requests can be mathematically embodied as:

$$\lambda = \frac{\left[R_1 + R_2 + ... + R_n\right]}{T} \qquad (1)$$

These requests are processed at each of the mentioned servers (web server, application server and database server) and is being denoted by μ.

$$\mu = \frac{\left[Rp_1 + Rp_2 + ... + Rp_n\right]}{T} \qquad (2)$$

Based on equations (1) and (2), there are three possible conditions, on which the 3-Tier e-Business architecture can be based.

Condition 1: When $\lambda > \mu$, this case is often referred as Transient State. If $\lambda > \mu$, then, there will be overflow of data of each servers. This makes the system unstable. Hence, under no circumstances λ should be greater than μ.

Condition 2: When $\lambda = \mu$, this case is called as Null State. This is a very typical case and randomly occurs. This state is typical used for academic studies only, and, practically, this neither occurs nor is desirable.

Condition 3: When $\lambda < \mu$, this case is termed as Ergodic State. If this situation is maintained then there will be finite queue length of data, which will be needed to be stored at stated server.

The anticipated Job Scheduling Algorithm is based on Condition 3 and uses the GI/G/3/n/k queuing model stipulations, for investigation.

3. Mathematical Analysis for the Proposed Job Scheduling Algorithm-Algorithms of Various Distributions

The biggest challenge in designing the job scheduling algorithm is that the requests at the system arrives and departs in a random manner. So, the prevalent confront in that under this status also, the system should toil in ergodic comportment for constancy of the architecture. It must also be eminent that the requests which disembark at the web server and the responses which are generated either from the application server or database server through application server do not pursue any distribution. Thus, the name "General Distribution" is being specified to the arrival of requests and departure of responses from the system and is being denoted by "GI" and "G" correspondingly.

Queue Analysis is necessary to provide the service time for each requests to make the architecture free from overflow of data. The analytical calculation for GI/G/3/n/k queuing model is quite complicated. Thus, the comprehensive study simulation is being conducted in the proposed architecture. Two different strings for 15000 requests and responses are to be stored in the system. They are represented as Ar, Ar(i)(for $i \epsilon 0 \text{ to } 15000)$ and Dr, Dr(i)$(i \epsilon 0 \text{ to } 15000)$ for Requests and Responses respectively. These two Strings are used to compute the Queue Length for the implementation of e-Business architecture in the Cloud Computing Environment. The Queue Length describes temporarily staying of the requests in the memory.

For simplicity, General distribution is amalgamation of five dissimilar types of distribution, specifically:

- Equiprobable distribution
- Gaussian distribution
- Geometric distribution
- Bernoulli distribution
- Exponential distribution

The comprehensive experimentation is being conducted for 15000 requests at a given unit of time. It is also being considered that the 3000 requests are owed to each of the above-mentioned distribution concurrently and the responses are generated consequently.

3.1 Algorithm for Gaussian distribution

Algorithm for Gaussian distribution is given below:

```
begin
read n
for i=1 to n in step of one do
for j=1 to n in step of one do
sum=0
P(j)=RAND U(j)
sum=sum+P(j)
y(i) = 1/12*sum
Write(y(i))
end for
end for
end
```

3.2 Algorithm for Equiprobable distribution

The Algorithm to generate equiprobable distribution is given as under:

```
begin
for i=1 to 3000 in step of one do
RAND [U_i]
Write[U_i]
end for
end
```

The algorithm generates 3000 Equiprobable numbers and is stored in an array.

3.3 Algorithm for Negative-Exponential distribution

The algorithm to generate 3000 negative exponential distributions using equiprobable distribution is given as under:

```
begin
read b
for i=1 to 3000 in step of one do
```

x(i) = RAND U(i)
y(i) = -1/b log(1-x(i))
Write(y(i))
end for
end

3.4 Algorithm for modified Geometric distribution

The Algorithm of modified Geometric distribution is given as under:
begin
read λ,n
b=1/λ
a=1-b
delta=b/a
for i=1 to n in step of one do
x(i)=RAND U[i]
*y(i)=(1-exp(delta*x(i))/b*
Write(y(i))
end for
end

3.5 Algorithm for Bernoulli distribution

The Algorithm of Bernoulli Distribution is as follows:
begin
read λ,n
b=1/ λ
a=1-b
for i=1 to n in step of one do
x(i)=RAND U(i)
*y(i)=(-a+SQRT(a*a+2*b*x(i))/b*
Write(y(i))
end for
end

4. Flowcharts for General Distributions of GI/G/3/n/k Queuing Model

There are three flowcharts that are to be used for comprehensive investigative study of the e-Business architecture in a Cloud computing milieu. First algorithm portrays the arrival process in the scheme. Since, General distribution does not follow any distributions, permutation of five different types of distributions are to be used in the study. Each distribution is allocated 3000 requests depending upon the availability of the request channel. These requests are merged by Web Server through Job Pooling (See Figure 8) to be processed by the Application Server and Database Server respectively. As the requests get processed by the system, the response generated forms another queue, which is also General distribution. Third flowchart depicts the service time evaluation methodology for the system.

4.1 Flowchart for generation of General Distribution of Arrival of Request (Ar(i))

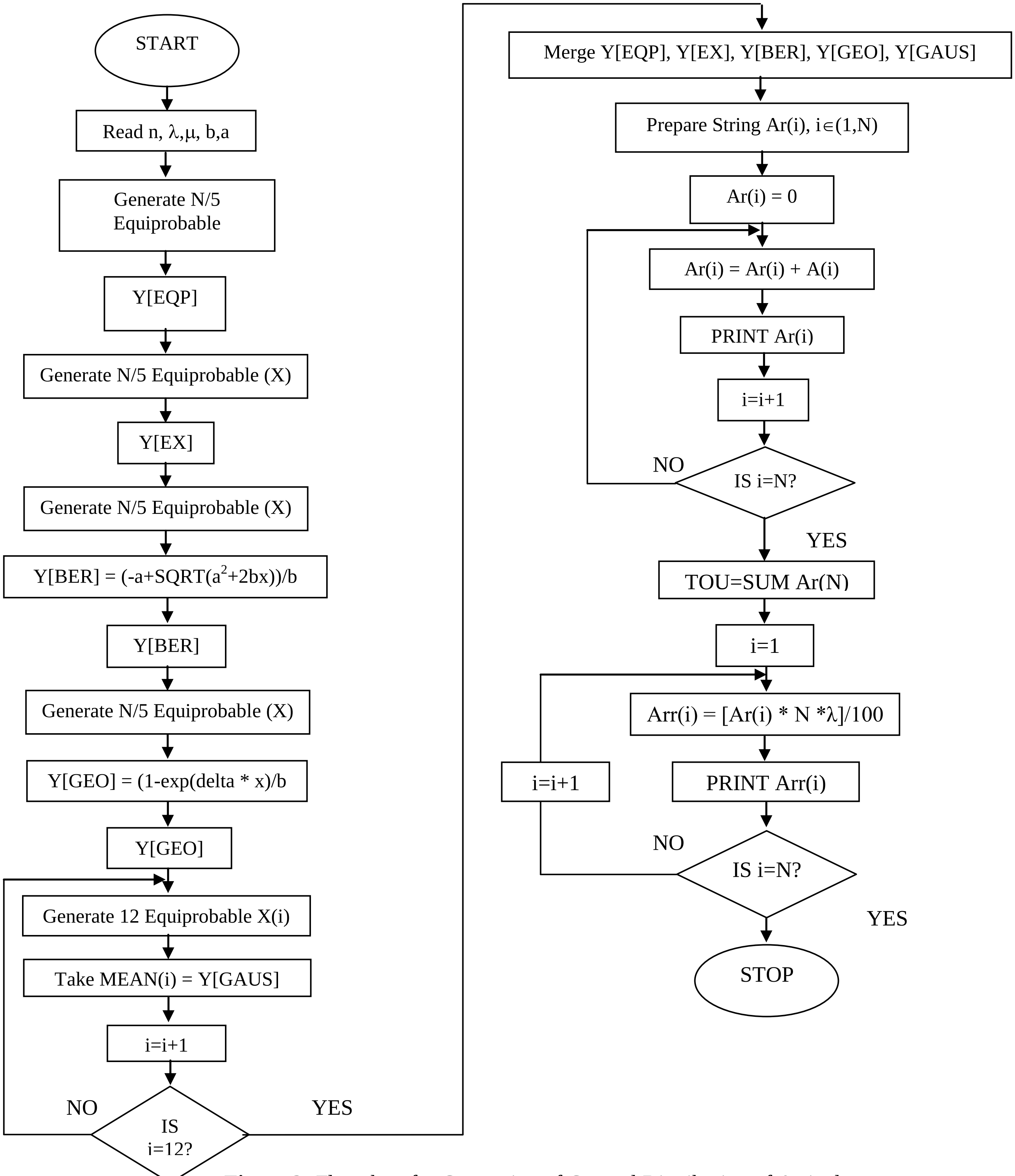

Figure 3: Flowchart for Generation of General Distribution of Arrival

4.2 Flowchart for Response Process having General Distribution with condition {Ar(i) <Dr(i); i ∈(1,N)

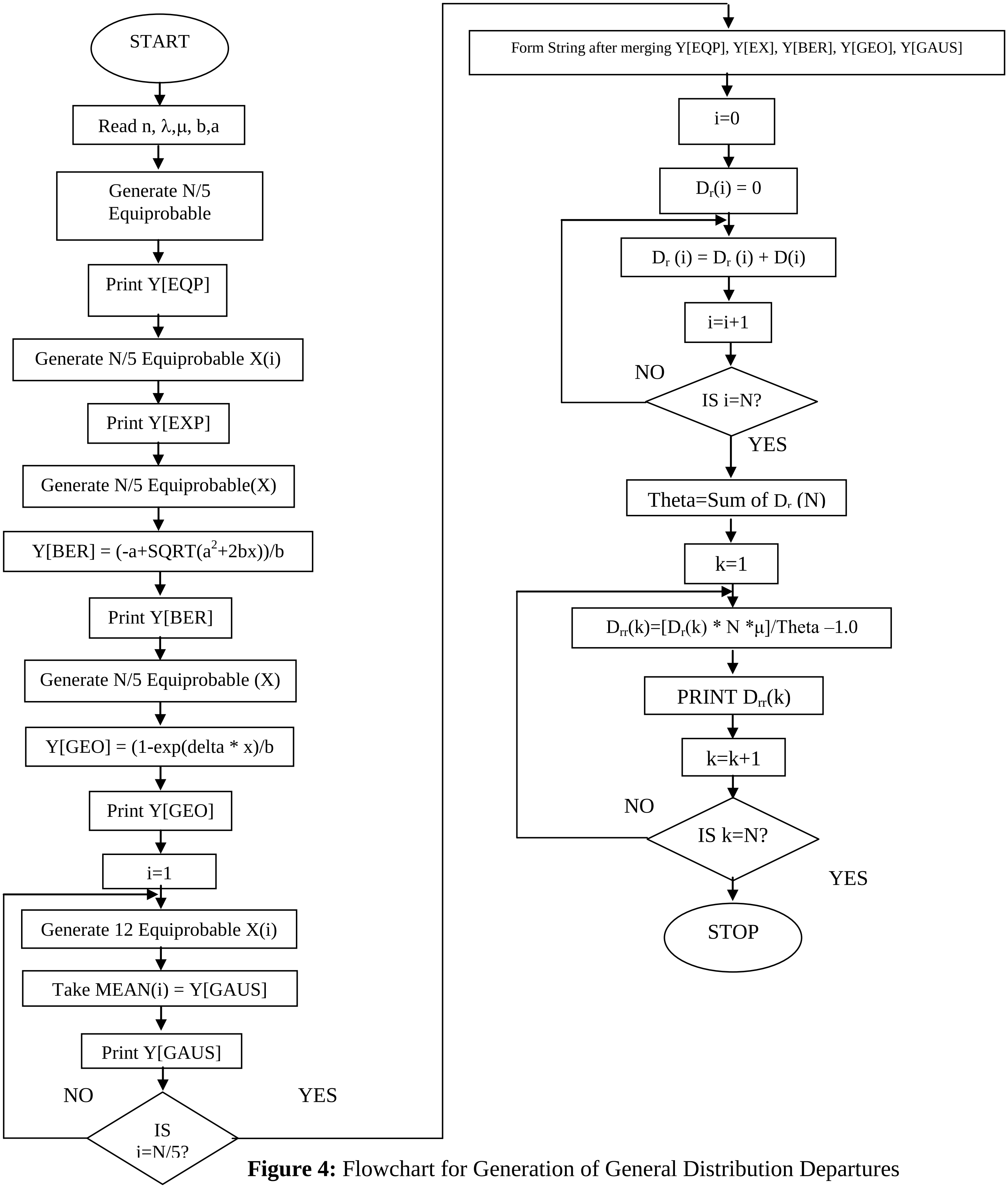

Figure 4: Flowchart for Generation of General Distribution Departures

4.3 Flowchart for Computation of Service Time in GI/G/3/n/k queue model

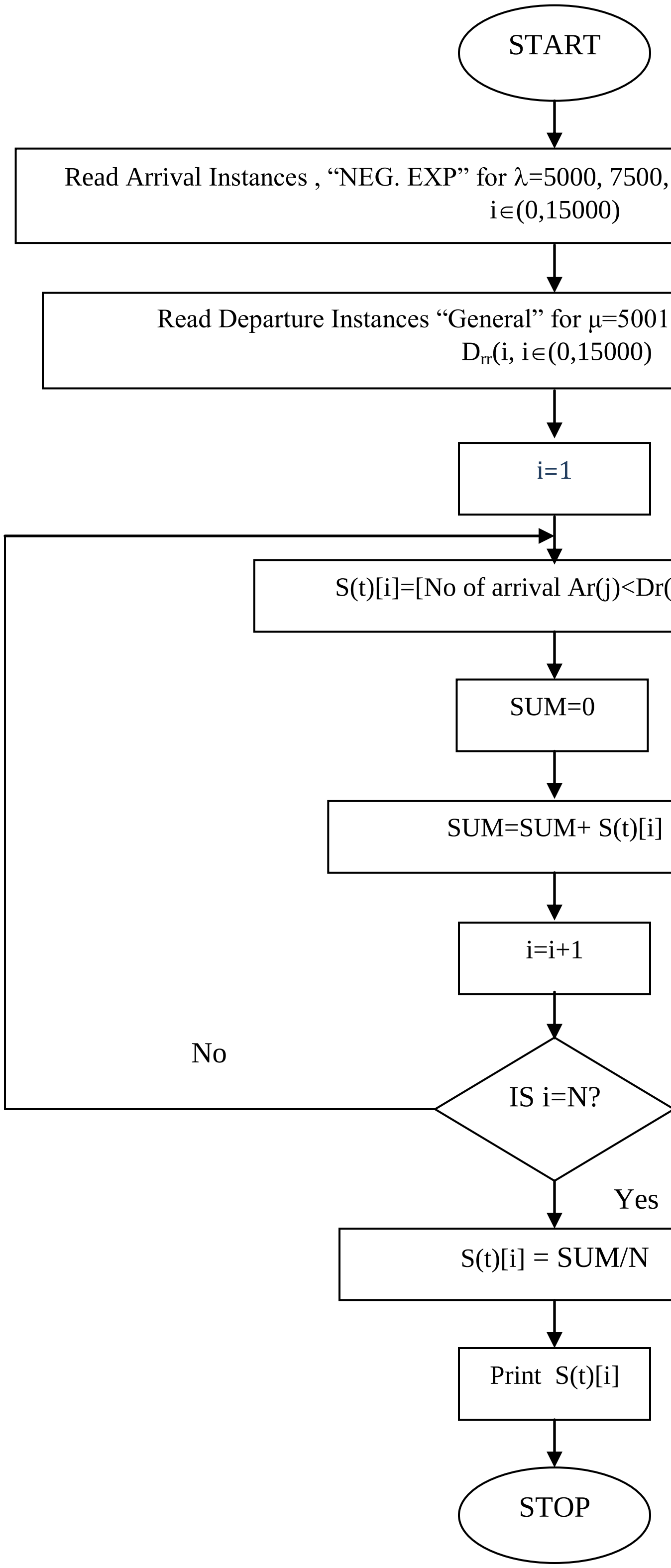

Figure 5: Computation of Service time in GI/G/3/n/k queuing model

It will be General distribution arrival or General distribution departure needs to be studied here, so that, two sets of Strings of size 15000 be stored in the System. They can be called, $Ar, Ar(i)(for\ i \in 0\ to\ 15000)$ and $Dr, Dr(i)(for\ i \in 0\ to\ 15000)$. These two Strings are used to compute the overall Service time S(t) by the proposed architecture.

5. Evaluation of Job Scheduling Algorithm via GI/G/3/n/k algorithm in a Client Server Environment

In cases of simulation study, best results are obtained when entire program are seen at one stretch of time. As random generation is used to be pseudo, hence, manipulation should be carried out for the generation of random number. The results obtained after running the programs are consolidated in the Figure 6.

λ	μ	Response generated by the System (Service time/s) [M/M/1 Queuing model]	Response generated by the System (Service time/s) [GI/G/3/n/k system]
5000	5001	5000	5538
7500	7501	7500	8400
10000	10001	10000	11190
12500	12501	12500	14126
15000	15001	15000	17695

Figure 6: Service Time Computations

Based on the results obtained by executing the Job Scheduling algorithm in a Client Server environment, we obtain that Service time S(t) for M/M/1 queuing model, which is a linear equation with a slope of unity. For GI/G/3/n/k, if we observe the plot, based on Figure 6, we obtain curves. The curve obtained is smooth and can be assumed to be a second order polynomial, which is close to first order polynomial of M/M/1 Queuing model. The graphs between M/M/1 and GI/G/3/n/k queuing models is given in the Figure 7.

6. e-Business architecture implementation in a Cloud Computing environment

The proposed e-Business architecture is an extension to the system of Client Server Computing. In the architecture, the Application Server and the Database Server are implemented in the public cloud. So the customer using the application is not fretful with the complexities of the Business logic and is offered the complete web application with added service. Thus, the architecture offers a significant workload shift. In the proposed architecture, the Web server has to no longer do the entire profound lifting when it comes to running applications. The network of clouds, which includes Application Server and Database Server(s), hold the impediments of the architecture. Also, the hardware and software demands on the user's side dwindle and the web server only executes the architectures interface software, whilst the cloud network takes care of the rest. The comprehensive architecture is depicted in Figure 8.

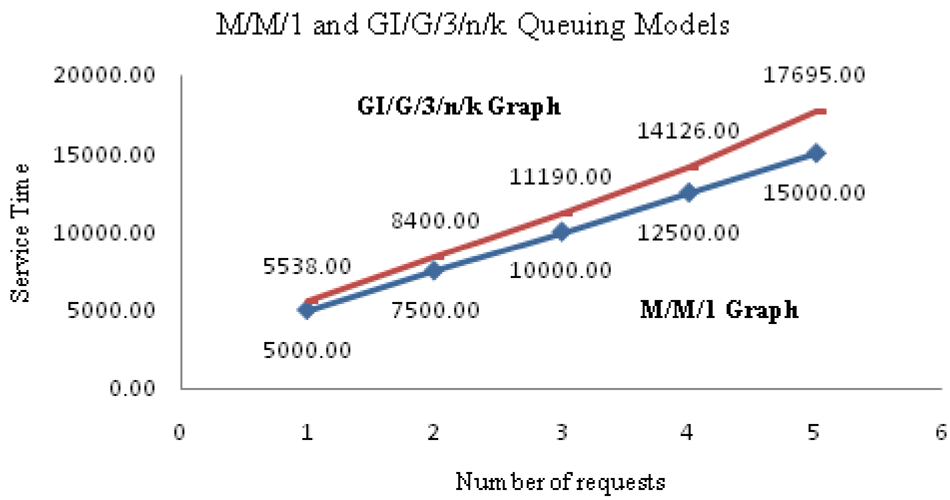

Figure 7: Graph between M/M/1 and GI/G/3/n/k queuing models

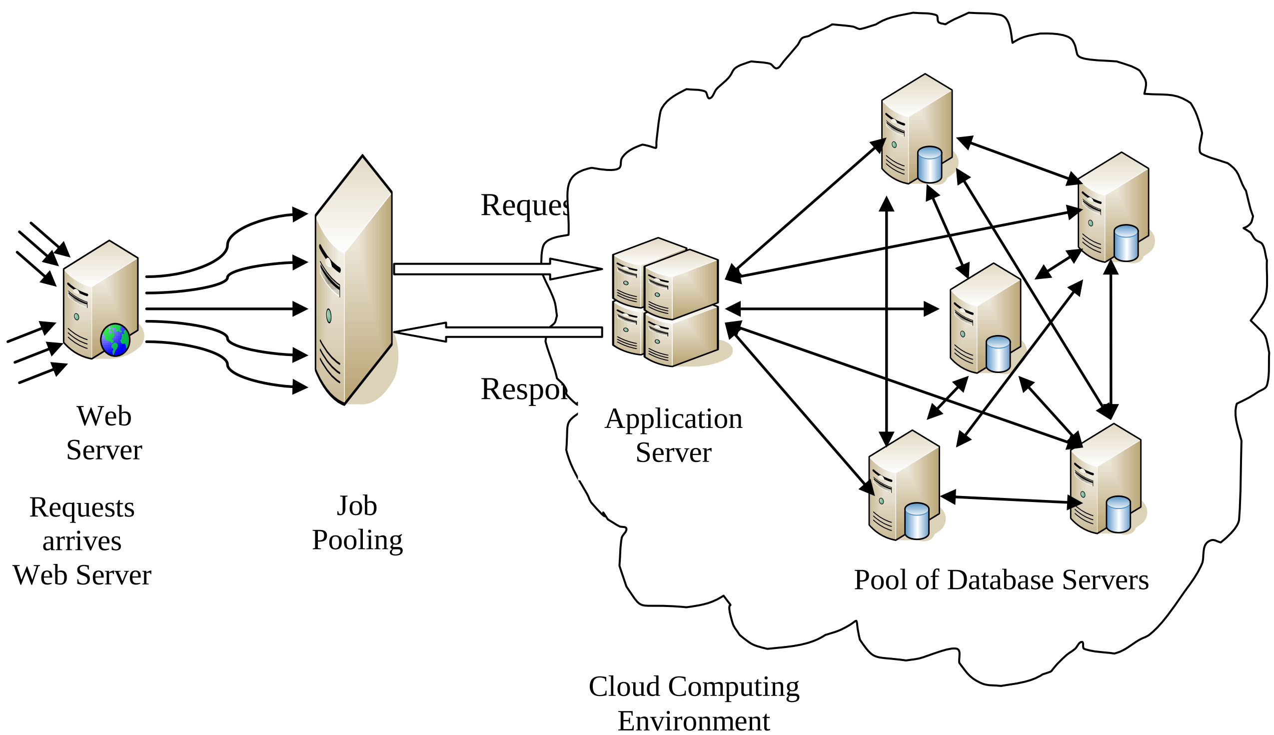

Figure 8: Implementation of e-Business architecture in a Cloud Computing environment

6.1 Step by Step architecture explanation

As illustrated in Figure 8, all the clients requests is being received at the Web Server. Since, there is an adoption of General Independent distribution; the Web Server transfers the requests to one of the distribution channels. Since, five different distributions are used in the proposed architecture, which is shown using five straight lines in the figure (Figure 8). Job Pooling Server executes the software code for the arrival instances (flowchart given in Figure 3) and merges all the requests into one queue and transfers the requests to the Cloud Computing environment. In the cloud computing environment, the requests are being received by Application Server on FCFS basis and generate the business logic. Through the pool of database servers, the Application Server generates the response by using the departure instance software program (flowchart given in Figure 4). By the time, the cloud computing environment generates the responses, the Job Pooling system, generates another queue to the executed. Upon receiving the response, Job Pooling system executes the software program to evaluate the service time, *S(t)*.

In order to evaluate the service time, a mathematical study is being conducted by using the *"curve-fitting technique"*, which is also known as Regression Technique. The complete mathematical evaluation is taken in the next section.

7. Evaluation of Service Time in the e-Business architecture using Regression/Curve Fitting technique in a Cloud computing environment

Based on observation from Figure 7, the equation for service time for the GI/G/3/n/k queuing model can be assumed as:

$$S(t) = a_2\lambda^2 + a_1\lambda + a_0 \qquad (3)$$

What is required now, is to calculate the values of a_0,a_1 and a_2 for the best fitted curve which has minimal error. To have minimal error in regression, mean square error is made minimal to give good result.

It is assumed at x_i represents the rate of arrival of the requests on the web server in the e-Business architecture, then, the value of Service time, will be represented by y_i is represented by

$$y_i = a_2{x_i}^2 + a_1x_i + a_0 \qquad (4)$$

where, a_0,a_1 and a_2 are coefficients of polynomial for GI/G/3/n/k model.

It must be also noted that, equation (6) is valid for larger rates of arrival of requests, which includes CASUAL EFFECT. This cannot be employed for lower rates of arrival.

If "S" represents the error in computation and real values of Service time, then, "S", which is square of derivation, is mathematically represented as:

$$S = \Sigma(y_i - \overline{{y_i}^2}) = \Sigma(y_i - a_2{x_i}^2 - a_1x_i - a_0)^2 \qquad (5)$$

If we differentiate S w.r.t., a_0,a_1,a_2 and setting each of these coefficients equal to zero, we get

$$na_0 + a_1\sum x_i + a_2\sum {x_i}^2 = \sum y_i \qquad (6)$$

y_i ↓ x_i →	5000	7500	10000	12500	15000	$\sum x_i = 50000$
y_i	5538	8400	11190	14126	17695	$\sum yx_i = 56959$
x_i^2	25x10^5	5625x10^4	10^8	15625x10^4	225x10 x10^6	$\sum x_i^2 = 5625 \times 10^5$
x_i^3	125x10^9	421875x10^6	10^{12}	1953125x10^6	3375 x10^8	$\sum x_i^3 = 6875 \times 10^9$
x_i^4	625x10^{12}	31640625x10^8	10^{16}	244140625x10^8	50625x10^{12}	$\sum x_i^4 = 88828125 \times 10^9$
$x_i y_i$	2769x10^4	63x10^6	1119x10^5	176575x10^3	265425 x10^3	$\sum x_i y_i = 64459 \times 10^4$
$x_i^2 y_i$	13845x10^7	4725x10^8	1119x10^9	22071875x10^5	3981375 x10^6	$\sum x_i^2 y_i = 79185125 \times 10^5$

Figure 9: Variables for GI/G/3/n/k queuing model

Figure 9 is being constructed to the calculation of coefficients of polynomial.

$$a_0 \sum x_i + a_1 \sum x_i^2 + a_2 \sum x_i^3 = \sum x_i y_i \qquad (7)$$

$$a_0 \sum x_i^2 + a_1 \sum x_i^3 + a_2 \sum x_i^4 = \sum x_i^2 y_i \qquad (8)$$

where, "n" represents the degree of polynomials as "n" equations are formed for the summation. Equations (6), (7) and (8) are three linear equations in three unknowns. These are called normal equations for quadratic regression and can be solved by Gaussian Elimination technique. The last column of Table 2 are substituted in the normal equations (as envisaged in equations (6), (7) and (8)), the following set of equations are obtained:

$$5a_0 + 50000a_1 + 562500000a_2 = 56949$$
$$50000a_0 + 562500000a_1 + 6875000000000a_2 = 7918512500000$$
$$562500000a_0 + 6875000000000a_1 + 88828125000000000a_2 = 7918512500000$$

By Gaussian elimination and using back substitution, the values of a_0, a_1 and a_2 are as follows:

$a_0 = 2.009\ X\ 10^{-7}$
$a_1 = 0.845028571$
$a_2 = 1.7828571\ X\ 10^{-5}$

Based on equation (3), the mathematical equation for Service time is:

$$S(t) = 1.7828571\ X\ 10^{-5} \lambda^2 + 0.845028571\lambda + 2.009\ X\ 10^{-7} \qquad (9)$$

From equation (9), the service time computations for the job scheduling algorithm in a Cloud Computing environment is given in Figure 10

λ	Response generated by the System (Service time/s) in a Cloud Computing environment
5000	4670.857
7500	7340.571
10000	10233.14
12500	13348.57
15000	16686.86

Figure 10: Service Time Computations

8. Performance analysis

The preeminent slant for the performance analysis is to weigh against the outcome obtained in Figure 10 and bout with the results of Figure 6 and formulate elucidations accordingly.

λ	Service time of the proposed algorithm in Client Server environment	Service time of the proposed algorithm in Cloud Computing environment
5000	5538	4670.857
7500	8400	7340.571
10000	11190	10233.14
12500	14126	13348.57
15000	17695	16686.86

Figure 11: Comparative analysis of the results

The results obtained are depicted by means of graph for an effortless performance investigation in Figure 12.

9. Conclusion

The study proposed the Job scheduling algorithm based on GI/G/3/n/k queuing representation. The algorithm was implemented in an e-Business architecture both in Client Server environment and Cloud computing setting. The study demonstrates that the algorithm gives the enhanced consequences in a Cloud computing scenario. The result was based on 15000 requests being received per unit time, wherein, each of the distribution channels was allocated 3000 requests apiece. The algorithm was based on ergodic condition and permanence was maintained and scrutinized throughout the implementation of the experimentation. The study is premeditated to be demeanor even advance, when elevated amount of requests being acknowledged and Service time being assessed accordingly.

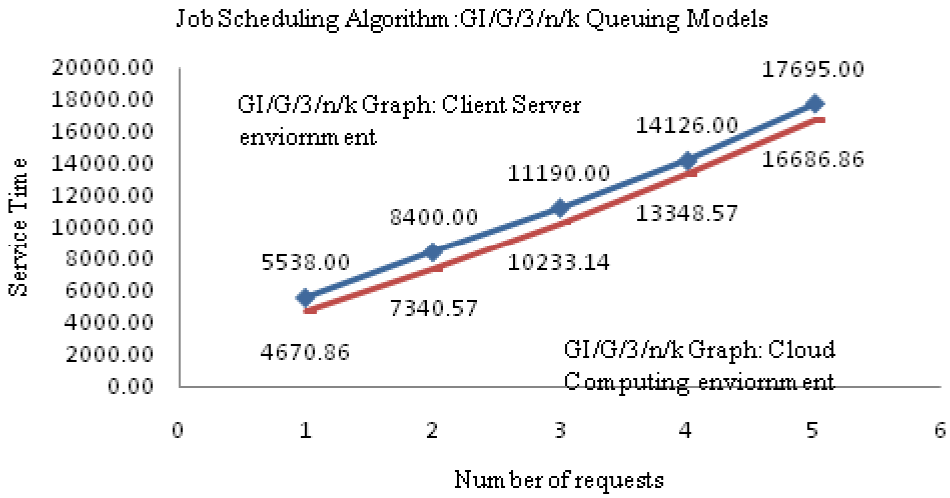

Figure 12: Graphical representation of the Job scheduling algorithm in both the environment

Figure 12 clearly depicts that the proposed Job scheduling algorithm takes quite a less service time in Cloud computing environment than on the Client Server environment, which proves the effectiveness and efficiency of the proposed algorithm.

References

[1] http://www.fub.it

[2] http://www.internetworldstats.com/stats.htm

[3] http://www.thbs.com/pdfs/Cloud-Computing-Overview.pdf

[4] Riktesh Srivastava et.al, "Design and Implementation of G/G/1 Queuing Model Algorithm w.r.t. its Applicability for Internet Gateway Server", International Arab Journal of Information Technology (IAJIT), Volume 5, Number. 4, October 2008, ISSN: 1683-3198.

[5] Riktesh Srivastava, et.al, "Memory Estimation of Internet Server using Queuing Theory: Comparative Study between M/G/1, G/M/1 and G/G/1 Queuing Model", International Journal of Computer and Information Science and Engineering (IJCISE), Volume 1 Number 2, 2007 ISSN: 1307-4164.

[6] Riktesh Srivastava, et. al, "Estimation of Buffer Size of Internet Gateway Server via G/M/1 Queuing Model", International Journal of Applied Science, Engineering and Technology, Volume 19, pp. 474-482, January 2007.

[7] Riktesh Srivastava, et. al, "Estimation of Buffer Size of Internet Gateway Server via G/M/1 Queuing Model", International Journal of Applied Science, Engineering and Technology, (IJASET) Volume4, Number 1, 2007, ISSN: 1307-4318.